Building in-silico Pathway SBML models from heterogeneous sources

I. Kanaris, K. Moutselos, A. Chatziioannou, I. Maglogiannis *Member IEEE* and F. N. Kolisis

***Abstract*— The recent revolutionary developments concerning the high throughput (-omics) measuring techniques in life sciences is expediting the way for the development of in silico models envisioning the systems biology perspective in the description of biological problems. As a result, very large open biological databases provide in silico descriptions in various formats, of biochemical pathways related to various cellular physiological aspects across the evolutionary climax. However, the lack of standardization regarding conceptual biological data representation incurs sheer limitations with respect to the functionality as well as the scientific completeness of the respective models. In this work, a software solution is presented which successfully bridges the gap towards building in-silico metabolic pathway models in Systems Biology Markup Language (SBML) format (standard SBML, CellDesigner SBML) by exploiting various XML based formats (SBML, KGML- KEGG Markup Language-, CellML - Cell Markup Language-, for pathway representation). Our solution provides methods for the biochemically correct transformation, curation and automatic simulation of the pathways, thus accomplishing the setup of fully functional in-silico models.**

I. Introduction

BIOINFORMATICS today is gaining impetus, gradually evolving to a ripe independent scientific field, rather than a logistic supporting service to the experimentalist. The advent of the high throughput measuring techniques (-omics era) in the field of life sciences, has radically reshaped the way research is conducted by providing a deluge of functional information regarding the interplay among genes, proteins, metabolites and whole cellular pathways, which processing requires proper annotation. As a result, large repositories of biological information regarding various different fields have been set along with tools and software capable of manipulating them. Unfortunately, the omnipresent lack of standardization, regarding the protocols and formats adopted for biological information conceptualization on behalf of the various database initiatives poses a sheer compatibility problem. The same problem arises in the case of modeling putative cellular biochemical reaction networks (metabolic pathways, signaling cascades, protein translational events). As a result, a multitude of biological databases offer information in many different ways, following their own proprietary formats for representation, which are not easily used, combined, inter-converted or cross-checked. In order to overcome these obstacles, the task of integration amid various tools and platforms, using distributed databases and markup languages to make models easily comparable and even cooperative, is becoming imperative. At present, the vast majority of annotated pathways are described in these three formats, namely SBML, CellML and KGML. All three of them constitute XML representations which describe metabolic pathways by defining compounds as nodes and reactions as edges, delineating complex graphs that are illustrated appropriately while in parallel for the case of SBML and CellML support simulating capabilities.

Currently the SBML (Systems Biology Markup Language) format, [1] has gained the widest acceptance concerning detailed representation of biochemical pathways, which has given rise to a wealth of SBML-complying editing visualization and simulation tools for this scope. A reliable repository of SBML converted detailed biochemical pathways is the BioModels Database of the European Bioinformatics Institute (http://www.ebi.ac.uk/biomodels/).

The CellML standard (which stands for Cell Markup Language) [2], also provides visualizing and simulation capabilities. Like SBML, it uses the MathML standard for kinetic law implementation that gives the models the ability to simulate reactions between compound elements. The CellML organization is also a very active community providing the end users many different tools for creating, visualizing and simulating pathway models. Its repository is also relatively small with only about 330 models in its Repository so far: (http://www.cellml.org/models).

Finally, the KEGG (Kyoto Encyclopedia of Genes and Genomes) database developed the KGML standard for visualizing the pathway models included in it (KEGG Markup Language, http://www.genome.jp/kegg/xml/). Being only visually oriented has no capabilities of importing MathML or other type of simulation related information and thus it is used only by the Kyoto Institute. The advantage of this particular standard is the huge database that it is based upon, including over 13000 metabolic models for over 150 different organisms.

In the following two paragraphs, a brief overview of related

A. C. and F. N. K. are with the Metabolic Engineering and Bioinformatics Group, Institute of Biological Research and Biotechnology, National Hellenic Research Foundation, Athens, Greece (authors e-mails: achatzi@eie.gr, corresponding author to provide phone: +30-210-7273759; fax: +30-210-7273758; e-mail: kolisis@eie.gr).

I. K. and K. M. are with the Department of Information and Communication Systems Engineering, University of the Aegean, Samos Greece (e-mails: kanaris.i@aegean.gr and kmoutselos@sch.gr).

I. M. was with Department of Information and Communication Systems Engineering, University of the Aegean, Samos Greece. He is now with the University of Central Greece, Faculty of Applied Sciences, Department of Informatics with Applications in Biomedicine (e-mail: imaglo@ucg.gr).

work will be given together with a short discussion about issues regarding the pathway modeling and conversion. Next, a detailed description of the proposed implemented approach will be presented closing with a general discussion and future work.

II. Related work

In order to overcome the problem of this diversity and to make all this distributed information about metabolic pathways available to everyone, tools specialized in parsing and transforming these standards have been developed. Based on the fact that SBML appears to be the dominant standard for pathway modeling in the future, we focus on creating a tool that can convert models from both CellML and KGML format as accurately as possible. In this section we will present some of the previous approaches regarding this effort.

In the case of CellML, the CellML2SBML tool was developed in 2004. Based on the fact that both file formats rely on XML standard, it uses the XSLT transformation technology to convert the CellML models to their SBML equivalents. This tool, based on a command line environment, incorporates a SAX XML parser that contiguously applies four different XSL style sheets to the input model resulting in a pure SBML model that includes all the visualization and simulation capabilities of the source file. Given the fact that both use MathML in defining kinetic laws the effort is mostly focused on resolving semantic and syntactic issues. As mentioned in [4], its capability of transforming CellML 1.1 files to SBML Level2 files reached 93% which is relatively high.

In the case of KEGG database, the tool KEGG2SBML [9] was developed in order to create SBML models by parsing KEGG data found in the LIGAND database. This tool was implemented using Perl programming language in 2004 and managed to transform accurately about 81% of the total metabolic pathway models available. This tool is based on parsing the data from flat files and is capable of exporting SBML models in both Level1 and Level2 format while also supporting CellDesigner's [5] annotation that retains the original visual layout of KEGG models. CellDesigner is a powerful tool for designing and simulating SBML pathway models, compliant with the Systems Biology Workbench (SBW) [11] with an easy-to-use graphical interface that provides distinct representation of various entity types through the use of metadata imported as annotations into standard SBML models.

III. Addressing issues

Examining the aforementioned approaches and their results, several issues arise regarding the successful steps towards building curated, functional and simulation ready in-silico pathway models for use in Systems Biology applications. While the CellML approach indicates that the problem is almost solved – given the fact that the resulting models are fully functional and originally designed with simulation purposes in mind – the issue of correctly transforming curated pathways of the KEGG database to real 'in-silico' models for simulation purposes is far from being resolved.

Two major issues arise when one refers to the transformation of KEGG metabolic pathways. First of all, KGML (KEGG Markup Language) an XML based format specific for data representation in the KEGG Database does not provide for kinetic information inclusion regarding the reactions it describes, thus providing just graphic illustrations of the pathways suitable only for visual representation. This poses a huge limitation to the effective utilization of this knowledge source, as in this form the files describing the pathways are inappropriate for simulation purposes. Furthermore, manually adding the kinetic laws in those models can be a time exhaustive effort, concerning the size of the pathways in terms of encompassed reactions and substrates. Therefore, an automatic method of identifying and importing appropriate kinetic mechanisms based on the type of each reaction is strongly advocated.

Another issue is the original design of the models. In order to have a clear layout and to help end users to examine them, many compounds and reactions are included more than once in the same model. This is again a major drawback regarding the simulation procedure, mainly from the biological point of view given the fact that all reactions take place in the same cell compartment and the multiple presences of compounds are definitely wrong. Furthermore, this way it is hard to define an overall concentration of an enzyme or compound that exists in the model thus gratuitously increasing the models complexity.

These problems are preserved in the KEGG2SBML approach, where the models created are simple mappings of the database to static SBML models.

IV. Our approach

In order to resolve these issues, we mostly focused on KGML models transformation following a different approach. The goal is to create accurate in-silico SBML versions of KEGG metabolic pathways, appropriate for dynamic simulations which at the same time provide a detailed realistic overview of the network. Furthermore, great effort is made towards the implementation of interfaces that can be easily used by end users with little or no previous experience in similar tools. In the following paragraphs the steps of converting the KGML pathway models to their SBML equivalents will be described in detail.

A. Model conversion

Given the fact that both formats are using the XML technology to represent their data, the most straightforward way manipulating them is by using XSL transformation. By applying an XSLT template on the source file, syntactic and semantic issues are easily resolved keeping all the original information to the resulting file.

In this step, the elements of <entry> group in KGML document are forming the <species> group in SBML format including compounds, genes or enzymes and neighboring pathways. Accordingly, information that resides in both <reaction> and <relation> nodes of the input file that specify how the compounds of the network interconnect with each other, create the <reaction> group of SBML. More specifically the reaction elements in KGML define the reactants and products while modifiers are defined in relation elements that link to specific enzymes in the entry group together with the anchor points of the neighboring pathways.

In order to provide further compatibility and enhanced functionality to the resulting models, two different XSL style sheets were implemented: one transforming models to pure SBML Level2 format and another that supports in addition the CellDesigner's annotation format. This particular annotation scheme has been selected due to its very good visualization information that it encapsulates and the very good environment of the CellDesigner tool regarding the simulation process.

B. Pathway Curation

After the initial step of XSL transformation, the resulting file is already compliant with SBML format and can be read and edited by every tool available for this platform. However, this is very difficult to be managed by a human researcher mostly due to the naming of entities. Keeping the source information, all the compounds of the model are using their KEGG ID naming making them unidentifiable by human readers. Furthermore, great redundancy usually is observed in these models due to the multiple presences of species and reactions.

In order to eliminate these problems the model is further processed using functions written in Java, implementing Xerces XML parser. The curation of the models follows two distinct stages:

1) Duplicate and orphan elimination

In this phase all the different compounds are scanned based on their KEGG ID names. If the same name is found in more than one species, they are all substituted by the first one found in the compound list. An example can be seen in the following Figures:

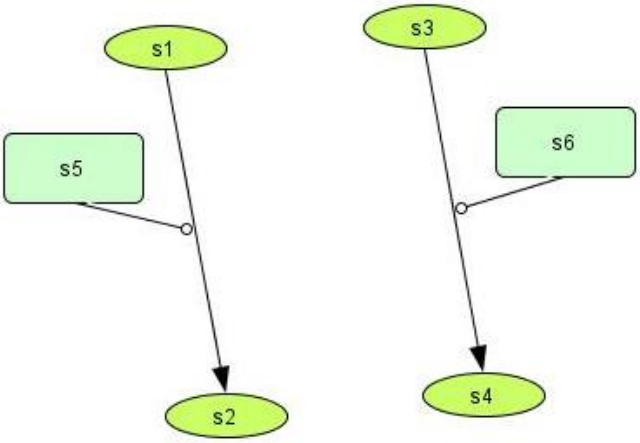

Fig. 1. Initial state of two reactions with the modifiers s5 and s6 having different IDs but representing the same entity.

In Figure 1 an example of two different reactions with one reactant, one product and one modifier, is presented. In the case that the modifier s5 is the same with the s6, the first will take the place of the second in its reaction (Figure 2).

The same applies also for duplicate reaction elements where all their attributes – such as name, reactants, products and modifiers – are equal. This usually happens in the case of combining more than one neighboring pathway models that have overlapping sections.

Following this substitution however, some of the initial compounds become nodes that do not connect to any reaction and together with other already "orphan" nodes from the initial model are deleted since they do not contribute to the reactions network in any way.

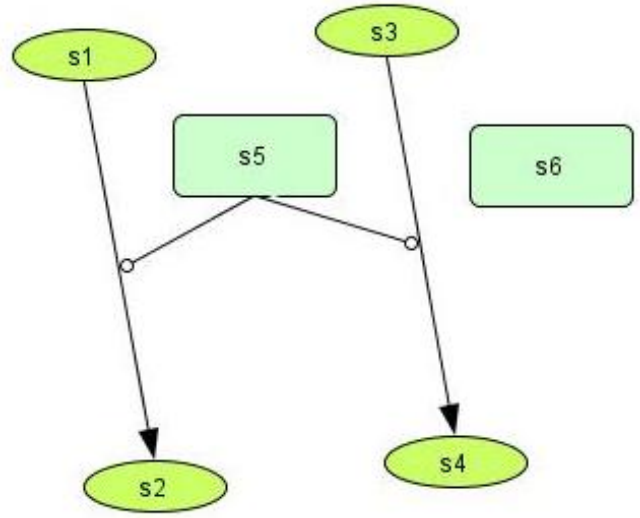

Fig. 2. Reactions after the substitution of duplicate element

2) Pathway annotation

After removing from the model all the unnecessary entries, the step of annotating each node according to the official KEGG database, is being followed. In order to do that, a set of flat files in tab delimited text format is being used. The approach of submitting queries directly to the dbget system of KEGG was not preferred, mostly due to speed issues. The time needed to parse a tab delimited file

and load it into memory is far less than this needed to submit queries through internet. Tab delimited files are four:

- *Compounds.tab;* Contains two columns: Compound IDs and their first name applied by the KEGG database (in the case they have multiple names)
- *EnzymeNames.tab;* Two columns again. The first has the International protein index (format 1.2.3.4) and the second has again the first name applied by KEGG database.
- *GlycanNames.tab;* Same as Compounds, first the Glycan IDs and second their first name by KEGG database.
- *Enzyme-Organism.tab;* This one has three columns. First the Enzyme index (same as in EnzymeNames.tab), then the KEGG Organism ID and third a Gene ID. This file is very helpful when managing Organism specific pathways.

As mentioned above, each compound is identified from its KEGG ID found in the original file. When a KGML file is converted to its SBML equivalent, every KEGG ID becomes the species name attribute. When annotating the model, the value of this attribute is being searched through the appropriate file for a match. In order to decide if the ID in question is a compound, enzyme, glycan or gene, the prefix of the ID is examined. If the name starts with the prefix "cpd:" it refers to a compound, "glycan:" is for a Glycan, "ec:" refers to Enzyme and if it has organism id followed by ":" it's a gene. For example "hsa:893" it's the gene with index 893 that belongs to human (homo sapiens = hsa). In reference pathways the reactions are catalyzed by enzymes, while in organism specific pathways the genes are shown as reaction modifiers. Using the Enzyme-Organism.tab we can easily correlate the organism gene with the Enzyme that it encodes and both are assigned as a name to the entity. In many cases though, an enzyme is encoded by more than one gene but only the first of them found in the database is included so to avoid long naming tags.

C. Deploying kinetics to the models

In order to complete the transformation of the originally imported pathways into fully functional simulating models, the identification and deploying of kinetics is crucial.

There are various software packages for simulations of biochemical reaction networks the following hold a prominent role: Gepasi[6], Copasi[7], CellDesigner, Smart Cell[12] and the SBW platform tools such as JDesigner and Jarnac. Among these, Gepasi provides a detailed kinetic library with various kinetic laws for description of the reaction rate type. In addition to the predefined kinetic types it also offers the possibility of user-defined kinetic types. Furthermore, new equations can be added to the existing database for future use. Gepasi library of predefined kinetics consists of well-known rate law expressions taken from published studies, which are the result of in-vitro experiments. Whenever a user opens a ready-made model, the program recognizes the equation by its name, and so the kinetic type is identified immediately. This feature is lost in the case of SBML exporting, due to the loss of extra information and the user is unable to know if a kinetic is one of the known and documented ones or some altered version.

Copasi tool on the other hand goes the identification one step further. When a model is exported to SBML, the kinetic reaction is saved in a <functionDefinition> block using MathML. During the process of importing back a previously created model, the tool identifies these blocks as patterns easily recognizable and thus retrieving the name of a known kinetic. However if the contents of the tag are slightly changed this recognition fails and a generic name is applied. Same as in Gepasi, the name of the kinetic type is irretrievable by other SBML simulation tools.

The approach of enriching the KEGG models with kinetics presented in this paper can provide a Gepasi-like functionality in deploying and recognizing kinetics. In the following paragraphs two different implementations of the deployment of kinetics will be presented.

1) Enhancing the SBMLeditor tool

In order to check the validity of the proposed methodology and to verify the quality of the resulting models the first implementation was based on SBMLeditor [8], an open source Java software distributed under the GNU General Public License. SBMLeditor is a 'low level editor of SBML files where users can create and remove all the necessary bits and pieces of SBML in a controlled way, that maintains the validity of the final SBML file'. Using the libSBML API [10] it manages to give a concrete environment that prevents erroneous entries while in the same time providing a very easy-to-use method for kinetics deployment.

The first steps of transformation – XSL transformation, curation and naming – are included as simple import options in the File menu. User can choose to import both pure SBML and CellDesigner's annotated models by selecting the options 'KGML2SBML' and 'KGML2CD' respectively. Also the ability of importing CellML documents using the set of four contiguously applied XSLT templates of the CellML2SBML tool has been implemented.

After finishing the transformation, the procedure of importing the kinetics into the model is human-aided. Using the already present feature of the tool, user can right click on a particular reaction and add a new Kinetic Law. Next by right-clicking on the blue 'kineticLaw' tag and select 'Edit' pops up the 'kineticLaw' window presented in Figure 3, where the user has the option to manually edit the MathML block. The new feature implemented is this: By clicking the 'Kinetic types' button, a window pops up. In it, user can see a list of kinetic laws proposed by the identification algorithm that match the form of the particular reaction, and especially the one highlighted as the most appropriate of all. The criteria used for this identification are:

1) Reversible / non-Reversible reaction

2) Number of parameters
3) The kinetic formula itself

In each step the possible equation set is reduced until the third one where a final identification is attempted using the exact formula pattern.

Additional information presented on the form is the ASCII form of the kinetic type and the name of parameters included in it together with their default values set by the algorithm. User can set them manually and by clicking 'OK' the import of the rate law is finished.

The specific implementation requires minimum effort by the end user, relieving him from the task of inserting the rate law without prior information and demanding only a little "fine tuning" of the parameters according to user's needs. Furthermore, the validation of the models based the existing architecture of SBMLeditor results in the production and exportation of very high quality models portable to every available tool supporting this format. One drawback however, is that in the case of transforming models to Celldesigner's annotated format it is very demanding in memory use, and the increase of maximum heap size used by Java virtual machine is mandatory.

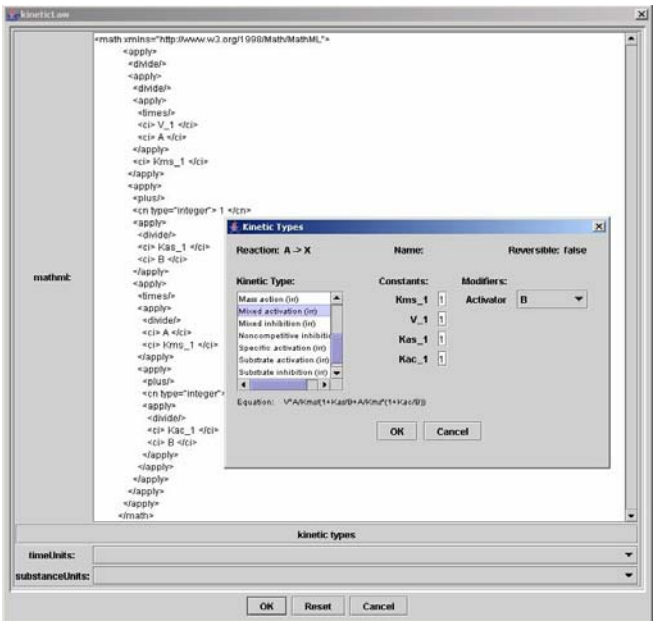

Fig. 3. Identification and customization of kinetics

2) KeggConverter application

Given the fact that KEGG database is a large metabolic pathway repository, a more 'massive' approach has to be built in order to test an overall success rate of the approach and of course to help users that need a large amount of models converted in a 'quick and easy' way. Taking into account also the large memory consuming of the SBMLeditor based approach, a command line tool was implemented (KeggConverter), which may be downloaded from (http://195.251.6.234/keggconverter/index.html).

This is a stand-alone application that provides the conversion abilities of our approach in a more massive and automated way. User can put the original KGML files to the 'in' folder and by giving the appropriate arguments to the tool he/she can retrieve the resulting SBML files from the 'out' folder. The available arguments are:

a. **'justConvert'** The application converts the KGML models to their equivalents using the pure SBML format without including any kinetic laws.
b. **'justConvertCD'** Converts the models to the CellDesigner's annotated format without importing kinetic laws.
c. **'makeKinetics'** This converts and introduce kinetics automatically in the produced pure SBML files.
d. **'makeKineticsCD'** Introduce kinetics in produced CellDesigner's annotated files.
e. **'STATS'** Exports statistics on resulting SBML models about the types of different reactions found in them.

In the cases of a. and b. the models are converted and curated as mentioned above. If the user decides to use the automatic kinetics deployment ability of the tool, every reaction found in the converted models will be identified and the appropriate kinetic will be inserted. The rate law addition is carried out according to the following rules:

- If a reaction is of 1-1-1 type (which means: 1 reactant, 1 product, 1 modifier) then the *hyperbolic modifier rate-law* is being added. If the reaction is reversible, the reversible hyperbolic modifier equation is being added.
- For ALL the rest types of reactions, the *mass action rate law* is being added (appropriately modified for the reversible or irreversible cases). If there are modifiers in the reaction, they remain in the SBML definition, although mass action doesn't have modifiers, so that no loss of information occurs. It is left to the user to decide later how to deal with each specific case.

This procedure of rate-law addition was determined after a statistic that was run on the whole set of KEGG SBML models, using the 'STATS' option of the tool. This option was developed as a simple utility function to help the control of the conversion process especially for massive KEGG models conversions. It searches in the 'out' subdirectory for existing SBML models, counts for all the found reactions the number of reactants, products and modifiers and outputs this information in two text files: "irrStatistics.txt" and "revStatistics.txt" that refer to irreversible and reversible reactions respectively.

Using this feature on our test case of converting the KEGG pathway database revealed that the vast majority of reactions found are of the type 1-1-1 with occurrence of 80% in irreversible and 90% in reversible reactions. This indicates that user has to do modifications on only about 10%-20% of the reactions that happen to be more complex in order to specifically define the functional parameters of it by changing the default mass action kinetic law applied automatically.

V. Discussion and Future Work

The task of building in-silico models for visualizing and simulating complex cellular functions such as metabolic

pathways is a very demanding job. From the collection of original expert knowledge about the processes taking place in a cell to the stage of fully functional representations of it in computer aided simulations, there are many drawbacks and obstacles that have to be overcome.

The approach presented in this paper is focused to the diffusion of existing knowledge about biochemical networks throughout the systems biology community. Based on XML representation format it manages to build well curated and fully functional models of biochemical networks by exploiting two major databases; CellML and KGML. Especially in the latter case, for the first time a deployment of kinetic rate laws is performed together with proper modifications making the models both descriptive and functional.

The development of two different kind of tools for both manually and automatic conversion and deployment gives the experimentalist the ability to work both massively and individually on the models. Algorithms for redundancy elimination and automatic kinetic law identification provide a great help in the effort of building in-silico models considering the large amount of data found in such networks.

Another issue that arises while building such models is the combination of two or more biochemical pathways, gradually reaching closer to full cell simulation capability. By performing some tests on KGML models, manually combining two or three pathway models in a single file and then converting it with our tool gave us very good results. Given the fact that the most important problem while combining separately created pathways is the existence of overlapping sections, the algorithms of pathway curation described above, solve the problem leaving only the needed compound and reactions. So, in future work we will try to automate this procedure also, giving the end user the capability to define multiple pathway combination and conversion taking as output models of larger scale.

Furthermore, the ability of automatic identification of kinetic laws will be enhanced, providing a more sophisticated system based on established knowledge about several reactions, including the definition of kinetic factors through the exploitation of open repositories such as the NIST Kinetics Database on the Web (http://kinetics.nist.gov) that provide that kind of knowledge.

REFERENCES

[1] M. Hucka, et.al., "The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models", Bioinformatics 19: 524-531.

[2] Cuellar, Autumn A., Lloyd, Catherine M., Nielsen, Poul F., Bullivant, David P., Nickerson, David P., Hunter, Peter J., "An Overview of CellML 1.1, a Biological Model Description Language", SIMULATION 2003 79: 740-747.

[3] Finney A, et. al., "Systems Biology Markup Language (SBML) Level 2: Structures and Facilities for Model Definitions", 2003; Available from: http://sbml.org/specifications/sbml-level-2/version-1/html/sbml-level-2.html.

[4] Schilstra, M.J., "Conversion of CellML 1.1 into SBML L2v1" 2004 http://sbml.org/software/cellml2sbml/conversion_of_CellML2SBML.pdf

[5] Akira Funahashi, Mineo Morohashi, Hiroaki Kitano, Naoki Tanimura, "CellDesigner: a process diagram editor for gene-regulatory and biochemical networks", BIOSILICO Volume 1, Issue 5, 5 November 2003, Pages 159-162.

[6] Mendes P., "Biochemistry by numbers: simulation of biochemical pathways with Gepasi 3", 1997, Trends Biochem Sci 22(9):361-3.

[7] Hoops S, Sahle S, Gauges R, Lee C, Pahle J, Simus N, Singhal M, Xu L, Mendes P, Kummer U., "COPASI - a COmplex PAthway SImulator", 2006, Bioinformatics.

[8] Rodriguez N, Donizelli M, Le Novère N., "SBMLeditor: effective creation of models in the Systems Biology Markup Language (SBML)", BMC Bioinformatics. 2007 Mar 6; 8:79.

[9] A. Funahashi and H. Kitano, “Converting KEGG DB to SBML,” Bioinformatics, 2003. 6.

[10] Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M., "LibSBML: An API Library for SBML", 2008, Bioinformatics, 24(6):880–881

[11] M. Hucka, A. Finney, H. M. Sauro H. Bolouri, J. Doyle, H. Kitano, "The ERATO Systems Biology Workbench: Enabling interaction and exchange between software tools for Computational Biology", 2002, Pacific Symposium on Biocomputing 7:450-461

[12] Ander, M. Beltrao, P. Di Ventura, B. Ferkinghoff-Borg, J. Foglierini, M. Kaplan, A. Lemerle, C. Tomas-Oliveira, I. Serrano, L., "SmartCell, a framework to simulate cellular processes that combines stochastic approximation with diffusion and localization: analysis of simple networks"